

PATENT ABSTRACTS OF JAPAN

(11)Publication number :2002-290744

(43)Date of publication of application :04.10.2002

(51)Int.Cl.

H04N 1/41  
H03M 7/40  
H03M 13/37  
H04N 1/415  
H04N 7/24

(21)Application number :2001-087375

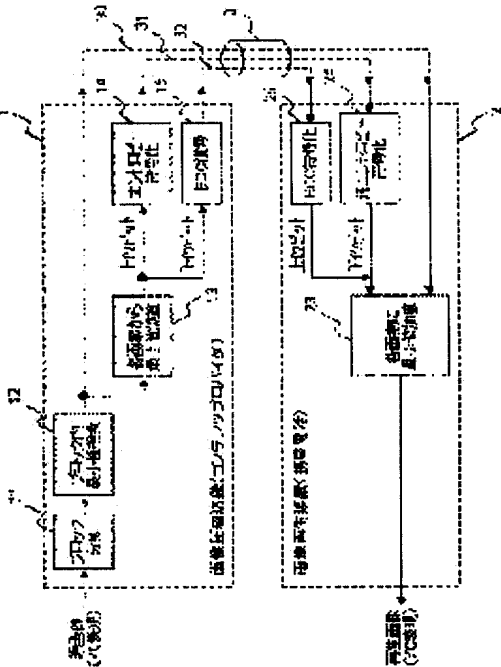
(71)Applicant :SYSTEM LSI KK

(22)Date of filing :26.03.2001

(72)Inventor :ESUMI ATSUSHI  
KOZUKI SEIJI  
NAKAURA KAZUHIRO

(54) IMAGE COMPRESSION SCHEME

(57)Abstract:  
PROBLEM TO BE SOLVED: To provide a new image compression scheme that provides as a high quality image as that provided by JPEG compression scheme with as small amount of calculation (calculation time) as with GIF compression scheme, thus allowing an apparatus having a limited calculation ability such as a portable telephone or the like to reproduce a high quality image.  
SOLUTION: Image compressing is performed by decode processing of error correcting codes, and image decompressing is performed by code processing of error correcting codes. Decode and code processing of error correcting codes may be performed using ordinary algorithm or referring to a table which is previously held as required. In a scheme for referring to the table, it is possible to correct a corresponding relation of the table so as to be able to perform optimum error correction as seen as pixel value. Entropy encoding can be used concurrently so as to increase compression rate. It is also possible to perform thinning out processing in bits of pixel value upon compression of image, and complementary processing upon decompression of image so as to further increase compression rate.



LEGAL STATUS

[Date of request for examination]  
[Date of sending the examiner's decision of rejection]  
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]  
[Date of final disposal for application]  
[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号  
特開2002-290744  
(P2002-290744A)

(43) 公開日 平成14年10月4日 (2002.10.4)

(51) Int.Cl. <sup>7</sup>	識別記号	F I	テコード* (参考)
H 0 4 N 1/41		H 0 4 N 1/41	C 5 C 0 5 9
H 0 3 M 7/40		H 0 3 M 7/40	5 C 0 7 8
13/37		13/37	5 J 0 6 4
H 0 4 N 1/415		H 0 4 N 1/415	5 J 0 6 5
7/24		7/13	A
審査請求 未請求 請求項の数17 O L (全 13 頁)			

(21) 出願番号 特願2001-87375(P2001-87375)

(22) 出願日 平成13年3月26日 (2001.3.26)

(71) 出願人 301016159

システムエルエスアイ株式会社  
愛媛県松山市北梅本町708番地3

(72) 発明者 江角 淳

愛媛県松山市鷹子町650番地1

(72) 発明者 上月 清司

愛媛県松山市北梅本町708番地3

(72) 発明者 中浦 一浩

愛媛県温泉郡川内町南方1883番地3

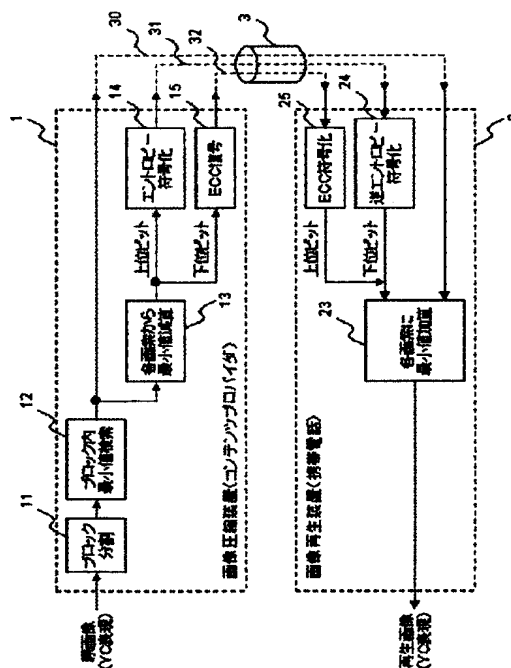
最終頁に続く

## (54) 【発明の名称】 画像圧縮方式

## (57) 【要約】

【課題】 現在知られている画像圧縮方式の代表例として、JPEG圧縮方式とGIF圧縮方式が挙げられる。携帯電話等の限られた演算しか行うことができないような機器への適用を考えたとき、JPEG圧縮方式は多くの実数乗算を伴い、膨大な処理時間を要する。また、GIF圧縮方式は自然画像を圧縮した際には画質が非常に劣化する、という問題がある。従って、計算量が少なく、高品質な画像が得られる圧縮方式は知られていない。

【解決手段】 画像圧縮時に誤り訂正符号の復号処理を行って圧縮を行い、画像伸張時に誤り訂正符号の符号化処理を行う。誤り訂正符号の復号、および符号化処理は、通常のアゴリズムを用いてもよいし、テーブルを保持しておき必要に応じて参照する方式でもよい。テーブルを参照する方式では、画素値としてみた場合に最適な誤り訂正を行えるようにテーブルの対応関係を修正することも可能である。圧縮率を高めるために、エントロピー符号化を併用することも可能である。更に圧縮率を高めるために、画像圧縮時に画素値のビット単位で間引き処理を行い、画像伸張時に補間処理を行うこともできる。



## 【特許請求の範囲】

【請求項1】静止画像に対する画像圧縮方式において、画像圧縮時に誤り訂正符号の復号処理を利用して圧縮を行い、画像再生時に誤り訂正符号の符号化処理を利用して伸張を行うことを特徴とする画像圧縮方式。

【請求項2】請求項1記載の画像圧縮方式において、誤り訂正符号の符号化処理を行って再生画像の画素値を得た後に、隣接画素の画素値と大小関係を比較し、必要に応じて誤り訂正符号で訂正し得る誤りパターンを加えることによって該画素の画素値と隣接画素の画素値との差異が小さくなるように画素値を修正することを特徴とする画像圧縮方式。

【請求項3】請求項1記載の画像圧縮方式において、誤り訂正符号の復号処理、および符号化処理に対応する写像テーブルを保持しておき、該テーブルを参照することで誤り訂正符号の復号処理、および符号化処理を実現することを特徴とする画像圧縮方式。

【請求項4】請求項3記載の画像圧縮方式において、原画像の画素値と再生画像の画素値の差異が最小になるように、請求項3記載の写像テーブルの対応関係を修正することを特徴とする画像圧縮方式。

【請求項5】請求項3記載の画像圧縮方式において、符号語ビット系列に輝度成分と色差成分が混在し、原画像の色差成分と再生画像の色差成分の差異よりも原画像の輝度成分と再生画像の輝度成分の差異が小さくなるように、請求項3記載の写像テーブルの対応関係を修正することを特徴とする画像圧縮方式。

【請求項6】請求項1記載の画像圧縮方式において、誤り訂正符号の復号処理、および符号化処理の適用部分を各画素値の下位ビットに限定し、上位ビットにはエントロピー符号化を用いることを特徴とする画像圧縮方式。

【請求項7】請求項6記載の画像圧縮方式において、各画素値と隣接する画素値の差分をとり、差分値に対して請求項6記載のエントロピー符号化を用いることを特徴とする画像圧縮方式。

【請求項8】請求項6記載の画像圧縮方式において、画像を複数の小ブロックに分割し、各ブロック内の画素値の最小画素値を検索し、各画素値と画素が属するブロックの最小画素値の差分をとり、差分値に対して請求項6記載のエントロピー符号化を用いることを特徴とする画像圧縮方式。

【請求項9】画像を構成する各画素の画素値を表現するために複数ビットからなるビット列を必要とする静止画像に対する画像圧縮方式において、画像圧縮時に、各画素の画素値を表現している複数ビットからなるビット列の全ビットを保持しておくか、あるいは該ビット列の全ビットではないいくつかのビットを保持しておくか、あるいは該ビット列の全ビットを保持しないかを、画素毎に決定することを特徴とする画像圧縮方式。

【請求項10】請求項9記載の画像圧縮方式において、

画像を複数の小ブロックに分割し、画素値を表現している複数ビットからなるビット列の全ビットを保持しておくか、あるいは該ビット列の全ビットではないいくつかのビットを保持しておくか、あるいは該ビット列の全ビットを保持しないかを各ブロックごとに一定のパターンで決定し、各ブロック内には該ビット列の全ビットを保持しておく画素が少なくとも1画素は含まれていることを特徴とする画像圧縮方式。

【請求項11】請求項10記載の画像圧縮方式において、画素値を表現している複数ビットからなるビット列の全ビットではないいくつかのビットを保持しておいた画素の補間処理に関して、補間後の画素値を該ビット列の全ビットを保持しておいた画素の画素値と同じにするか、保持しておいたいくつかのビットから決定するかを、ブロック内に少なくとも1画素は含まれている該ビット列の全ビットを保持しておいた画素との関係に応じて切り替えることを特徴とする画像圧縮方式。

【請求項12】請求項1記載の画像圧縮方式、および、請求項9記載の画像圧縮方式を組み合わせた画像圧縮方式。

【請求項13】請求項1あるいは請求項9あるいは請求項12記載の画像圧縮方式を連続的に用いた動画圧縮方式。

【請求項14】請求項1あるいは請求項9あるいは請求項12記載の画像圧縮方式による画像圧縮あるいは画像伸張を実現するソフトウェアを格納した媒体。

【請求項15】請求項13記載の動画圧縮方式による動画圧縮あるいは動画伸張を実現するソフトウェアを格納した媒体。

【請求項16】請求項1あるいは請求項9あるいは請求項12記載の画像圧縮方式による画像圧縮あるいは画像伸張を実現するハードウェア。

【請求項17】請求項13記載の動画圧縮方式による動画圧縮あるいは動画伸張を実現するハードウェア。

【発明の詳細な説明】

【0001】

【発明が属する技術分野】本発明は、静止画像の圧縮方式に関するものである。

【0002】

【従来の技術】画像（ここでは、静止画像）を構成する最小単位を画素（pixel）といい、各画素は複数ビット（但し、2値画像の場合は1ビット）で画素の明るさを示している。この各画素の明るさを示す値を画素値という。例えば、各画素に8ビットずつ割り当てられていると各画素は256段階の明るさを表現することができる。

【0003】1枚の画像を表現するためには、画像サイズをaビット×bビット（以下では、このような画像をa×b画素と表記する）、各画素値がcビットで表現されているとすると、a×b×cビットの容量が必要となる。これは、モノクロ画像の場合であって、カラー画像の場合は、光の3原色であるR（red）、G（green）、およびB

(blue) の3色の明るさをもって表現する必要があるもので、 $a \times b \times c \times 3$ ビットの容量が必要となる。従って、画像の容量を $Co$ ビット（モノクロ画像の場合は $Co = a \times b \times c$ 、カラー画像の場合は $Co = a \times b \times c \times 3$ ）とすると、このような画像を伝送帯域  $B$ [bps] のインターネット等の通信路を介して伝送する際には、 $Co/B$ [sec]だけの伝送時間が必要である。また、記録メディアに記録する際には、 $Co$ ビットの記録容量が必要となる。

【0004】通常は、上述のように $Co$ ビットの容量の情報をそのまま伝送、あるいは記録するのではなく、より短時間で伝送するために、あるいは限られた容量の記録メディアにより多くの画像を記録するために、圧縮して伝送、あるいは記録されることが多い。圧縮される前の画像を原画像、圧縮された後の画像を圧縮画像、圧縮画像を元の画像に復元する処理を伸張、伸張された後の画像を再生画像という。

【0005】画像の圧縮は、圧縮画像を伸張したときに、再生画像として原画像と全く同じ画像が再現できる無歪圧縮と、原画像と全く同じ画像には再現できない有歪圧縮に大別される。一般に無歪圧縮よりも有歪圧縮の方が圧縮画像の容量（サイズ）は小さくなる。圧縮の結果、 $Co$ ビットの容量が $Cc$ ビットになったとき、そのアルゴリズムは $Cc/Co$ の圧縮率であるといい、 $Cc/Co$ の値が小さいほど圧縮率が高いという。

【0006】ところで、画像の表現方法としては、上述の $R$ 成分、 $G$ 成分、および $B$ 成分による表現（以下では、 $RGB$ 表現）の他、画像の輝度情報（ $Y$ 成分）と色差情報（ $C_r$ 成分、および $C_b$ 成分）で表現する方法（以下では、輝度・色差表現）が知られている。 $RGB$ 表現では、特にどの色が重要であるということではなく、各成分の重要度は全く同じであるが、輝度・色差表現の場合には、輝度情報には人間の視覚的に重要な情報が多く含まれているが、色差情報には重要な情報があまり含まれない、という特徴がある。従って、輝度・色差表現には、画像圧縮を行う場合に、色差情報の欠落は、再生画像の画質にさほど影響を与えないことを利用して、色差情報を大幅に圧縮できる、という利点がある。実際のテレビジョン放送で使用されている $YIQ$ 表現も輝度情報（ $Y$ 成分）と色差情報（ $I$ 成分、および $Q$ 成分）で表現されており、輝度・色差表現の一つである。

【0007】現在知られている多くの圧縮方法は、隣接する画素値には強い相関があることを利用して圧縮している。現在、具体的な画像圧縮（有歪圧縮）方式としては、JPEG (Joint Photographic coding Experts Group) 圧縮方式とGIF (Graphics Interchange Format) 圧縮方式が広く用いられている。以下に、これら2つの圧縮方式について、簡単に説明する。

【0008】(JPEG圧縮方式) まず、JPEG圧縮方式について説明する。JPEG圧縮方式に関しては、例えば“画像圧縮技術（越智 宏、黒田英夫著・日本実業出版

社)”に詳細に記述されている。

【0009】JPEG圧縮方式では、最初に画像を $8 \times 8$ 画素のブロックに分割し、各ブロックに対してDCT (Discrete Cosine Transform: 離散コサイン変換) を実行して、周波数領域の値に変換する。次に得られた周波数領域の値を量子化するが、この量子化の際に、高周波成分に対しては粗い量子化を行って大幅に情報を落とすが、低周波成分は細かな量子化を行ってあまり情報を落とさないようにする。最後に量子化を行った各成分に対してエントロピー符号化を行って、JPEG圧縮系列を得る。

【0010】JPEG圧縮方式は、人間の視覚が低周波成分に対しては敏感だが、高周波成分に対しては鈍感であることを利用して、上述の量子化を行う際に量子化のレベルを可変にして高周波成分を大幅に圧縮している。また、本来画像には高周波成分よりも低周波成分の方が多く含まれているため、高周波成分に対して粗い量子化を行うとほとんどの成分が“0”あるいは“0”付近の値になるため、エントロピー符号化を行った際に大幅な圧縮を行うことができる。

【0011】JPEG圧縮方式は、高圧縮率の割には優れた画質が得られる圧縮方式として知られている。また、圧縮率・画質をある程度自由に設定でき、目的に応じた圧縮率・画質を選択することもできる。一般には、圧縮率は1/5から1/50程度で用いられることが多い。

【0012】(GIF圧縮方式) 次に、GIF圧縮方式について説明する。GIF圧縮方式に関しては、例えば“Graphics Interchange Format Version 89a (modified) Programming Reference, CompuServe Incorporated”に記述されている。

【0013】GIF圧縮方式では、使用する色のテーブル（カラーパレットともいう）を保持しており、このカラーテーブルには、各8ビットで $RGB$ を表現した合計24ビットで表現される色（約1600万色）の中から任意の256色を用いることができる。実際の圧縮処理は、まず、原画像の画素値をカラーテーブル（256色）中の一つの色に対応させる。次に、カラーテーブルへの指標の系列を無歪圧縮方式の一つであるLZW (Lempel Ziv Welch) 法を用いて無歪圧縮を行う。

【0014】GIF圧縮方式では、圧縮率は1/4から1/6程度となる。

【0015】

【発明が解決しようとする課題】 上述のようにJPEG圧縮方式は、高圧縮率の割に高品質な画質が得られる方式として知られている。しかしながら、JPEG圧縮方式の問題点として、計算量が多く、圧縮・伸張処理に時間を要する点が挙げられる。PC等の比較的高速・高機能な装置間で通信を行う場合には計算量の問題はさほど重要ではないが、コンテンツプロバイダから画像表示可能な携帯電話への画像配信サービス等を考えた場合には大きな問題となる。このようなサービスにおいては、画像の圧縮は

プロバイダ側の処理になるため、計算量が多くなっても問題とはならないが、画像の再生は携帯電話で行うため、再生処理は高速に行える必要がある。すなわち、携帯電話では小型軽量化が重視されるため、必ずしも高性能な処理装置を搭載できるとは限らないためである。

【0016】また、GIF圧縮方式は、実数演算を伴わないので演算量は少なく、短い処理時間で圧縮・再生処理を行え、携帯電話等でも十分に再生が可能であるが、GIF圧縮方式の問題点として、再生画像がカラーテーブルの色、すなわち256色でしか表現されない点が挙げられる。パナーやアニメ画像等の人工的なグラフィックス画像を圧縮する場合には、これらの画像は使用する色が自然画像に比べて格段に少なく、また文字や直線を多く含む、という特徴があるため、GIF圧縮方式が適した圧縮方式といえる。しかし、人物写真、あるいは風景写真等の自然画像を圧縮する際には、約1600万色で構成されている画像の色を256色だけで表現しなければならず、この減色処理により、例えば、グラデーションになっている部分に縞状の段差ができやすい。また、単なる減色処理ではなく、ディザ法と呼ばれる手法を用いることもできるが、この方法では粒状感が目立った画像となってしまう。これら縞状の段差や粒状感は、人間の視覚にも敏感に反応し、画質を大きく劣化させる原因となる。

【0017】本発明の目的は、GIF圧縮方式と同程度の少ない演算量（計算時間）で、JPEG圧縮方式と同程度の高品質な画像が得られる新しい圧縮方式を提案することである。本発明により、携帯電話等の限られた演算しか行うことができないような機器でも高品質な画像を再生することができるようになる。

【0018】

【課題を解決するための手段】上述の目的を達成するために、本発明では、誤り訂正符号（Error Correcting Codes: ECC）の概念を用いている。通常の誤り訂正符号は、通信路上で通信を行う際に、kビットの情報系列に送信側で一定の規則に従った冗長系列を付加してnビットの符号語系列を作成し、受信側では付加された冗長分を利用して通信途中に生じた誤りを検出、あるいは訂正し、kビットの復号系列（推定情報系列）を得る、という使用法で用いられる。このような情報系列がkビット、符号語系列がnビットである符号を、一般に(n,k)符号という。

【0019】本発明では、画像を圧縮する際に、各画素値を受信側で受信された系列（情報系列+冗長系列）と考え、復号操作（ここでは、誤り検出・訂正+冗長部分の削除のことをいう）を行い、誤り訂正符号の情報系列（kビット）のみを保持しておく。例えば、誤り訂正符号として、(n,k)符号を用いれば、圧縮率は $k/n$ となる。圧縮画像を伸張する際には、誤り訂正符号の符号化操作を行う。すなわち、kビットの系列に誤り訂正符号の冗長ビットを付加して、nビットの系列を得る。

【0020】以上の処理を、画素値が7ビットで表現されている画像に(7,4)ハミング符号を用いた例を、図3を用いて説明する。図3は、原画像の画素値が"46"であった場合の処理を示している。7ビット系列"0101110"をあるハミング符号で誤り訂正を行うと、"0101100"となる。この系列の情報系列は"0101"、冗長部分は"100"であるので、保持しておく系列は"0101"のみであり、これが受信側に伝送される。受信側では、"0101"を符号化し、再生画像の画素値"0101100"を得る。

【0021】ここで、t重誤り訂正符号を用いているとすると、原画像の系列と再生画像の系列では最大でtビットが異なっている。tビットの誤りは、系列を画素値としてみたときにどの重みで生じるかを特定することはできないが、隣接画素の画素値には強い相関がある、という特徴を利用して修正を行うことができる。すなわち、重みが大きい誤りが生じている場合には、隣接画素の画素値と大きく異なり、誤りの大きさを推定することが可能となる。一方、誤りの重みが小さい場合には、再生画像に与える影響が小さいので、修正を行う必要がない。

【0022】ところで、画素値をcビットで表している場合、cと誤り訂正符号の符号長のnが、 $n=c$ の関係にあるときはそのまま適用することができるが、 $n \neq c$ の場合は必ずしも一致している必要はない。 $n > c$ の場合は、隣接する複数の画素に対してまとめて復号・符号化を行うことで対応できる。また、 $n < c$ の場合は、画素のビットを複数の符号語に分割して復号・符号化を行えばよい。

【0023】また、より高品質な再生画像を得るために、本発明では、上述の誤り訂正符号をテーブルとして保持し、原画像と再生画像の画素値の差分が最小になるように上述のテーブルを修正することもできる。誤り訂正符号の符号化、および復号処理は、kビット情報系列とnビット符号語系列の写像を実現すればよいので、その一つ的手段としてテーブルによる写像を行う。

【0024】テーブルによって写像を実現することの利点は、任意の対応関係の修正を行っても全く問題はない、という点である。例えば、(7,4)ハミング符号を例にとって説明する。(7,4)ハミング符号は1重誤り訂正符号であるので、最大1ビットの誤りが生じる可能性がある。すなわち、7ビットの系列を画素値としてみた場合には重み64、32、16、8、4、2、1の誤りが生じているか、あるいは誤りは生じていないか、のいずれかである。このとき、1つの誤りで重み64の誤りが生じるよりも、2つの誤りで重み3の誤りが生じる方が、再生画像の画質に与える影響ははるかに少ない。

【0025】一般的には、t個の誤りが生じた場合よりもt+1個以上の誤りがあったとしても画素値としての重みが小さい写像を行う方が、再生画像の画質に与える影響は少ない。従って、本発明では、上述のように原画像と再生画像の画素値の差分が最小になるように上述のテ

ーブルを修正する。なお、このときは生じる誤りの重みが小さくなるように修正しているので、上述のような隣接画素の画素値から誤りの大きさを推定して修正する処理は不要である。

【0026】以上の処理を、画素値が7ビットで表現されている画像に(7,4)ハミング符号のテーブルを用いた例を、図4を用いて説明する。図4の上段左のテーブルは通常のハミング符号の復号アルゴリズムを用いた場合の圧縮側のテーブル、上段右のテーブルは通常のハミング符号の符号化アルゴリズムを用いた場合の伸張側のテーブル、下段左のテーブルは修正後の圧縮側のテーブル、下段右のテーブルは修正後の伸張側のテーブルを示している。この例において、上段の通常のハミング符号テーブルでは重み64の1ビット誤りが生じているが、下段の修正後のハミング符号テーブルでは、重み5の2ビット誤りになっている。

【0027】また、より高い圧縮率を得るために、本発明では、上述の誤り訂正符号を適用する部分を画素値の下位ビットに限定し、上位ビットには従来から用いられているエントロピー符号化を適用する。これは、上位ビットほど隣接画素間の相関が強く、エントロピー符号化を行った際に高い圧縮率を得ることができるために、この処理を行う。誤り訂正符号の適用を下位Lビットに限定した場合、Lとnが必ずしも一致している必要はない。これは、上述のnとcの関係と全く同じ理由からである。

【0028】ここで、上位ビットには、そのままエントロピー符号化を用いず、隣の画素値との差分をとり、差分値に対してエントロピー符号化を行ってもよい。上述のように隣接画素間の画素値には強い相関があるため、“0”付近の値の出現確率が高くなり、エントロピー符号化を行った際に高い圧縮率を達成することができる。

【0029】また、上述のエントロピー符号化の圧縮効果を高めるために、本発明では、画像を複数の小ブロックに分割し、各ブロック内の画素値の最小値を検索し、各ブロック内の画素値から検索した最小値を減じる。本処理を行うことで、各ブロック内の本処理を行った後の画素値は、“0”または“0”付近の正の数になるため、エントロピー符号化を行った際に大幅に圧縮することができるようになる。

【0030】また、上述の目的を達成するために、本発明では、圧縮時に画素の間引き処理を行い、伸張時に画素の補間処理を行う。間引き・補間処理を行うことで圧縮率を高めることができ、伝送すべき情報量が減少する。間引き処理を行うとき、色差成分は輝度成分に比べて情報が欠落しても再生画像に与える影響は少ないので、この特徴を利用して間引き処理を行う。

【0031】具体的には、次の3パターンの間引きパターンを準備し、各画素値にいずれかの間引きパターンを割り当てる。

- ・ パターン2 . . . 全ビット保持している
- ・ パターン1 . . . 上位Uビットのみ保持している
- ・ パターン0 . . . 全ビット保持していない

上述のパターン0は、画素そのものを削除する、従来の間引き方法である。一方、パターン1は、本発明による新しい間引き方法で、画素単位ではなく、ビット単位（下位ビット）の間引き方法である。

【0032】間引きは、画像を複数の小ブロックに分割し、一定の間引きパターンを繰り返す。以下では、この小ブロックの間引きブロックと呼ぶことにする。各間引きブロックには情報を削減しない画素、すなわち間引きパターン2の画素が少なくとも1画素は含まれている必要があり、以下では、間引きパターン2の画素の一つの間引きブロック代表画素と呼ぶ。なお、間引きブロックは、上述のエントロピー符号化の圧縮率を高めるために使用した小ブロックとは、ブロックサイズやブロックの区切り方に関して、何ら関係がない。

【0033】上述の間引きパターンを用いて間引き処理を行った画像に対する伸張時の補間処理は、次のように行う。パターン2の画素は、全ビット保持しているため、補間の必要はなく、何ら処理は行わない。パターン1の画素は、Uビットの情報からcビットの情報を復元する必要がある。保持している上位Uビットの値が間引きブロック代表画素の上位Uビットの値と一致している場合には、ブロック代表画素の値をそのまま使用する。保持している上位Uビットの値が間引きブロック代表画素の上位Uビットの値と一致していない場合には、上位ビットには保持しておいたUビットの情報をそのまま使用し、下位ビットは、下位ビットの最上位ビットのみ“1”とし、残りのビットはすべて“0”とする。すなわち、下位ビットは“10 . . . 0”となる。なお、下位ビットは必ずしも上記のように決定する必要はなく、任意に決定してよい。パターン0の画素は、何も情報がないので、間引きブロック代表画素の値をそのまま使用する。

【0034】ここで、パターン1のビット単位の間引き処理を導入する効果について、図5、および図6を用いて説明する。図5、図6は、2×2画素の間引きブロックを使用した間引き・補間処理の一例であり、図5は、画素単位の間引き（パターン0）を用いた場合の図であり、図6は、ビット単位の間引き（パターン1）を用いた場合の図である。いずれも画像の画素値が急激に変化しているエッジ部分の処理を示している。エッジは、例えば人物画像の顔の輪郭、風景画像の建物の輪郭等、自然画像においては極めて重要な部分で、エッジが明確に再現されないとならば画質が大きく劣化する原因となる。

【0035】図5において左下のブロックに着目すると、間引きブロック代表画素60は領域Aの白い部分であるが、同じ間引きブロックに所属している画素61乃至63は、領域Bの灰色の部分である。しかし、間引き処理によって画素61乃至63の情報は完全に削除されてしまい、

再生時に間引きブロック代表画素60の情報がそのままコピーされてしまうので、画素61乃至63はすべて領域Aの画素として再現されてしまう。エッジ部分がある間引きブロックではすべて同様の現象が発生する。すなわち、ブロック状の歪が生じた画像になり、エッジは不明確になってしまう。

【0036】同じ左下のブロックについて、図6では次のようになる。図5の場合と異なり、上位ビットを保持しており、エッジのような画素値が大きく変化するような部分では、上位ビットまで異なっている場合が多く、このような画素では間引きブロック代表画素の画素値はコピーされない。従って、画素61乃至63は画素60の値がコピーされずに、保持しておいた上位ビットに、固定値"10...0"が付加されることになる。ここで、下位ビットは固定値となってしまうが、上位ビットまで異なっているような急激に画素値が変化している部分では、下位ビットの値はそれほど重要ではなく、再生画像の画質に与える影響は少ない。従って、図6のように本発明のビット単位の間引き（パターン1）を用いれば、再生画像のエッジはほぼ元通りに再現することができる。

【0037】パターンは画素毎に割り当てる、と記述したが、この割り当ては輝度成分、色差成分で同じ割り当てを行う必要はない。従って、上述のように、色差成分は、輝度成分に比べて再生画像の画質に与える影響が小さいことを利用して、色差成分は大きな圧縮率を得ることができるパターン0で画素単位で間引き、輝度成分は高品質な画像を得るためにパターン1でビット単位で間引き、というような間引きパターンを用いることもできる。

【0038】また、上述の誤り訂正符号と間引き（パターン1）・補間処理を混在させる場合には、誤り訂正符号を下位Lビットに適用し、上位H(=c-L)ビットはエントロピー符号化を用いて、間引きパターン1では上位Uビットだけを保持しておくものとする、このときのHとUは、 $H \geq U$ となっていることが望ましい。

【0039】また、間引き処理を行った場合にも、より高品質な再生画像を得るために、本発明では、上述の誤り訂正符号を実現する写像テーブルを以下のように修正する。すなわち、色差成分は輝度成分に比べて情報が欠落しても再生画像の画質に与える影響は少ないという特徴を利用して、nビット符号語系列として輝度成分、色差成分を混在させ、誤りを色差成分の方に多く生じさせるような修正を行う。この修正を行うことで、主に画質を決定している輝度成分の情報の欠落を減少することができる。

【0040】また、本発明は、適用範囲を静止画像に限定するものではなく、本発明を連続的に使用することで動画にも適用することができる。

【0041】さらに、本発明には、本発明による画像圧縮方式あるいは動画圧縮方式による圧縮あるいは伸張を

実現するソフトウェアを格納した媒体（記録媒体や通信媒体など）も含まれる。

【0042】また、本発明は、ソフトウェアとしてだけでなく、半導体などのハードウェアにより実現することも可能である。

【0043】

【実施例】本発明の具体的な実施例について、図1を用いて説明する（実施例1）。

【0044】図1は、コンテンツプロバイダから画像表示可能な携帯電話への画像配信サービスを考え、本発明を適用した画像圧縮装置（コンテンツプロバイダ）1の処理、および画像再生装置（携帯電話）2の処理を示している。本システムの入力である原画像、および本システムの出力である再生画像は、共にY成分、Cr成分、およびCb成分で表現（以下では、YC表現）されているものとする。また、画像サイズは120×120画素で、各画素は8ビットで表現されているものとする。

【0045】以下では、圧縮処理、および伸張処理の各処理について、それぞれの処理の内容を詳細に記述する。なお、以下の説明で用いている処理ブロックとは、図1に示した処理ブロックの符号を示している。

【0046】（画像圧縮処理）

（圧縮ステップ1） ブロック分割（処理ブロック11）

YC表現されている原画像を $n \times n$ 画素のブロックに分割する。本実施例では、 $4 \times 4$ 画素のブロックに分割する。従って、原画像は900個のブロックに分割される。

【0047】（圧縮ステップ2） ブロック内最小値の検索（処理ブロック12）  
次に、各ブロックにある16(=4×4)個の画素値の中から、最小値を検索する。

【0048】（圧縮ステップ3） ブロック内最小値の減算（処理ブロック13）  
各画素値から、処理ブロック12で検索されたブロック内最小値を減算する。この減算の結果、ブロック内の各画素値は、少なくとも1画素が"0"、残りの画素は絶対値が小さな正の値となる。

【0049】（圧縮ステップ4） 画素値の符号化（処理ブロック14、および15）

次に、処理ブロック13で得られた各画素値を上位Hビットと下位Lビット( $H+L=8$ )に分け、上位Hビットに対しては、エントロピー符号化を行う（処理ブロック14）。本実施例では、上位1ビットと下位7ビットに分け、エントロピー符号化には、ハフマン符号とランレングス符号を組み合わせて使用する。また、残りの下位7ビットに対しては、誤り訂正符号の復号を行う（処理ブロック15）。本実施例では、誤り訂正符号として(7,4)ハミング符号を考え、一般的な復号アルゴリズムを用いて、各画素の下位7ビット系列を4ビットの系列に圧縮する。

【0050】以上の処理の結果、送信される情報は、処

理ブロック12で得られたブロック内の画素値の最小値30（圧縮は行わず固定長系列）、処理ブロック14で得られた各画素の上位1ビットの圧縮系列31（可変長系列）、および処理ブロック15で得られた(7,4)ハミング符号の復号結果の情報系列に相当する系列32（圧縮は行わず固定長系列）の3種類である。これらの情報は、通信路（無線、または有線）3を介して画像再生装置2に送信される。

【0051】（画像伸張処理）

（伸張ステップ1） 差分画像の再生（処理ブロック24、および25）

画像再生装置2では、まず受信系列32を符号化して冗長を付加することにより伸張する（処理ブロック25）。圧縮時に(7,4)ハミング符号を使用したので、伸張時にも同じ符号を用いる。受信系列32を4ビット毎に区切り、一般的な符号化アルゴリズムを用いて7ビットのハミング符号の符号語に符号化（冗長部分を付加）する。(7,4)ハミング符号は1重誤り訂正符号であるので、符号化時に最大で1ビットの誤りが生じている。この1ビットの誤りを画素値としてみたとき、重みが64、32、16、8、4、2、および1の誤りが生じている場合、および誤りが生じていない場合の8通りが考えられる。このうち、例えば重みの大きな誤りに関しては、隣接画素の画素値と値が大きく離れることになるので、隣接画素との誤差をとり、誤差の大きさがある閾値以上である場合には、重みの大きな誤りを加える、あるいは減じることにより、誤差が小さくなるように修正を行うことが可能である。一方、重みの小さな誤りに関しては、絶対値が小さく再生画像に与える影響は少ないので、修正する必要はない。次に、系列31に対して、圧縮の際行った規則と同じ規則で逆エントロピー符号化を行う（処理ブロック24）。処理ブロック25で得られた7ビットの系列を下位ビット、処理ブロック24で得られた1ビットの系列を上位ビットとして接続して8ビットの系列とし、差分画像の画素値とする。

【0052】（伸張ステップ2） ブロック内最小値の加算（処理ブロック23）

差分画像の各画素値に対して、その画素が属する4×4ブロックの最小値情報30を加算し、再生画像の画素値とする。

【0053】 上述の実施例1では、YC表現された画像を原画像とした例を示したが、原画像がYIQ表現されている場合でも、輝度情報、色差情報という意味では全く同じであるのでそのまま適用することができる。また、例えばRGB表現のように、輝度情報と色差情報に分離されていないフォーマットの画像を扱う場合には、圧縮ステップ1（処理ブロック11）の前処理として、原画像フォーマットからYC表現に変換し、伸張ステップ2（処理ブロック23）の後処理として、YC表現から原画像フォーマットに逆変換すればよい。

【0054】 また、上述の実施例1では、圧縮ステップ1において4×4画素のブロックを用いたが、このブロックのサイズは任意のサイズでよい。ブロックサイズが小さくなると最小値を減算した後の各画素の上位4ビットの分散が小さくなるためエントロピー符号化の圧縮率を高めることができ、系列31のビット数を減少させることができるが、ブロック数が多くなるために、系列30のビット数は増加する。逆に、ブロックサイズを大きくするとブロック数が少なくなるので系列30のビット数は減少するが、最小値を減算した後の各画素の上位4ビットの分散が大きくなるためにエントロピー符号化の圧縮率が低下し、系列31のビット数が増加する。

【0055】 また、上述の実施例1では、誤り訂正符号として(7,4)ハミング符号を用いているが、他の誤り訂正符号でもよい。例えば、エントロピー符号化を画素の上位4ビットに対して適用し、隣接2画素分の下位4ビット、すなわち8ビットの系列を考え、ここに(8,4)拡大ハミング符号を適用することもできる。

【0056】 また、上述の実施例1では、誤り訂正符号の復号アルゴリズム、および符号化アルゴリズムを使用しているが、写像を表現するテーブルを用いてもよい。実施例1のハミング符号の場合、圧縮側では128個の7ビット系列を16種類の4ビット系列に写像するテーブルを、また伸張側では16個の4ビット系列を16種類の7ビット系列に写像するテーブルをそれぞれ保持しておく。このとき、誤りのビット数が多くなっても画素値としての誤りの重みが小さくなるように修正しておいてもよい。

【0057】 また、上述の実施例1では、輝度成分、色差成分を間引くことなくそのまま使用している。また、両成分を同じように扱っている。しかし、上述のように、色差成分は輝度成分に比べて情報が欠落しても再生画像に与える影響は少ない。この特徴を利用して圧縮率を高めることができる。実施例を、図2を用いて説明する（実施例2）。

【0058】 図2は、図1と同じくコンテンツプロバイダから画像表示可能な携帯電話への画像配信サービスを考え、本発明を適用した画像圧縮装置（コンテンツプロバイダ）1の処理、および画像再生装置（携帯電話）2の処理を示している。図1との相違点は、ブロック分割前にパターン0間引き処理41を、また処理ブロック13の後にパターン1間引き処理を行い、処理ブロック23の後にパターン0補間処理43を、また処理ブロック23の前にパターン1補間処理44を行っている点である。以下に、間引き処理41、42、および補間処理43、44について説明する。なお、本実施例では、エントロピー符号化（処理ブロック14）および誤り訂正復号（処理ブロック15）の前に、各差分値を上位4ビットと下位4ビットに分けるものとする。原画像、および再生画像のサイズ等のその他の諸条件は、実施例1と全く同じである。

【0059】（間引き処理）間引き処理41、および42に

ついて、図7を用いて説明する。図7は、間引き処理を行う際の各画素の間引きパターンを示しており、上側から順に、Y成分、Cr成分、およびCb成分の間引きパターンを示している。小さい四角が1画素を示しており、“2”と記述している画素70、80、および90は全ビット保持しておくパターン2の画素、“1”と記述している画素71乃至73は上位4ビットだけ保持しておくパターン1の画素、“0”と記述している画素81乃至83、91乃至93は、全ビットとも保持せず削除するパターン0の画素である。また、図中の太枠7乃至9が2×2画素の間引きブロックを示しており、画素70、80、および90がそれぞれ間引きブロック代表画素である。

【0060】まず、Y成分の圧縮について説明する。上述のように、Y成分にはパターン0の画素がないので、パターン0間引き処理41に関しては、何も行わない。ブロック分割（処理ブロック11）からブロック内最小値の減算（処理ブロック13）までは実施例1と全く同様に全ビットに対して処理を行う。次に、パターン1間引き処理42において、パターン1である画素71乃至73に関しては、下位4ビットを削除する。従って、このような画素に対しては誤り訂正復号15の処理はない。パターン2であった画素70に関しては、全ビット保持しておくので、実施例1に示した処理を行う。

【0061】次に、Cr成分、およびCb成分の圧縮について説明する。上述のように、Cr成分、およびCb成分にはパターン0の画素81乃至83、91乃至93が含まれている。パターン0間引き処理41において、パターン0の画素81乃至83、91乃至93は、完全に削除される。従って、図7に示した間引きパターンを適用していれば、120×120画素であった画像サイズが、60×60画素となる。以下の処理は、実施例1と全く同じである。ブロック分割（処理ブロック11）を行う際のブロックサイズは元のサイズ4×4画素のまま使用してもよいし、画像サイズが小さくなっていることを考慮して2×2画素のブロックサイズを用いてもよい。

【0062】最終的に保持している情報を、図8および図9の最左列に示している。図8の最左列は、図7の間引きパターンで間引かれた各画素のY成分の差分値の一例を示している。また、図9の最左列は、図7の間引きパターンで間引かれた各画素のCr成分およびCb成分の画素値の一例を示している。画素値の網掛け部分は保持していないビットである。

【0063】（補間処理）パターン1補間処理44について、図8を用いて説明する。図8は、Y成分の差分値に関しての補間の様子を示している。画素70は、全ビット保持していた画素であるので、補間処理は不要である。画素71は、保持していた上位4ビットが間引きブロック代表画素である画素70と同じであるので、画素70の値をそのまま代入する。画素72は、保持していた上位4ビットが間引きブロック代表画素である画素70と異なっている

ので、上位4ビットは保持していた情報“0010”をそのまま使用し、下位4ビットは“1000（固定値）”とする。画素73も同様に、保持していた上位4ビットが間引きブロック代表画素である画素70と異なっているので、上位4ビットは保持していた情報“0001”をそのまま使用し、下位4ビットは“1000（固定値）”とする。

【0064】次に、パターン0補間処理43について、図9を用いて説明する。図9は、Cr成分およびCb成分に関しての補間の様子を示している。画素80は、全ビット保持している画素であるので、補間処理は不要である。画素81乃至83に関しては、全く情報を保持していないので、ブロック代表画素である画素80の値をそのまま代入する。Cb成分に関しても同様に、画素90は補間処理が不要で、画素91乃至93は画素90の値をそのまま代入する。

【0065】以上の処理は、図7に示した間引きパターンを使用した場合の補間処理の例であり、間引きパターンによっては、Y成分でも全ビット保持していないパターンや、Cr成分、あるいはCb成分でも上位4ビットを保持しておくパターンも存在し得る。

【0066】また、上述の実施例2では、間引きブロックを2×2画素のブロックとして、図7に示した間引きパターンを用いたが、間引きブロックのサイズ、および間引きパターンはこの限りではない。

【0067】さらに、上述の実施例2では、各画素の差分値あるいは画素値を上位4ビット、下位4ビットに分けるとしたが、差分値あるいは画素値の上位ビット、下位ビットへの分け方はこの限りではない。

【0068】

【発明の効果】本発明によれば、GIF圧縮方式と同程度の少ない演算量（計算時間）で、JPEG圧縮方式と同程度の高品質な画像を得ることができる。従って、本発明を用いれば、携帯電話等の限られた演算しか行うことができないような機器でも高品質な画像を再生することが可能となる。

【図面の簡単な説明】

【図1】 本発明を適用した画像圧縮装置および画像再生装置の処理ブロック図（1）

【図2】 本発明を適用した画像圧縮装置および画像再生装置の処理ブロック図（2）

【図3】 誤り訂正符号を用いた圧縮処理、および伸張処理の一例を示した図

【図4】 誤り訂正符号の写像をテーブルで実現し、対応関係修正の一例を示した図

【図5】 従来の画素単位の間引き処理を行った場合の圧縮・伸張処理の例を示した図

【図6】 本発明のビット単位の間引き処理を行った場合の圧縮・伸張処理の例を示した図

【図7】 間引きブロック、および間引きパターンの概念図

【図8】 補間処理の一例を示した図（1）

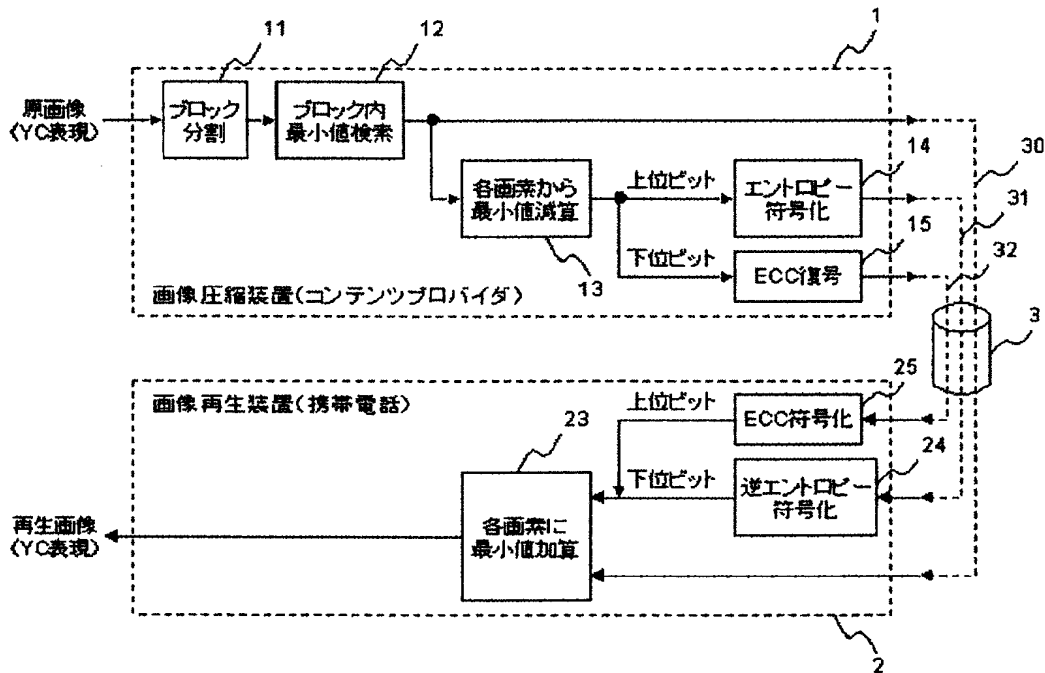
15  
【図9】 補間処理の一例を示した図 (2)

【符号の説明】

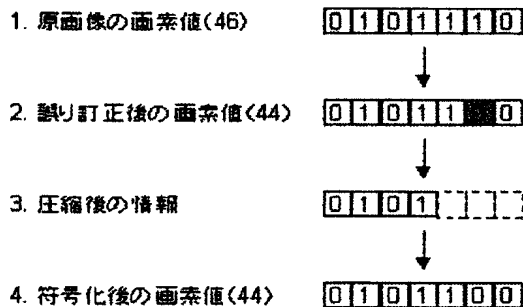
- 1 画像圧縮装置 (コンテンツプロバイダ)  
11~15 画像圧縮処理ブロック  
2 画像再生装置 (携帯電話)  
23~25 画像再生処理ブロック  
3 通信路 (無線または有線)  
30~32 送信する情報

- \* 41, 42 間引き処理ブロック  
43, 44 補間処理ブロック  
60 間引きブロック代表画素  
61~63 間引きブロック代表画素以外の画素  
7~9 間引きブロック  
70, 80, 90 間引きブロック代表画素  
71~73, 81~83, 91~93 間引きブロック代表画素以外の画素  
\* 画素以外の画素

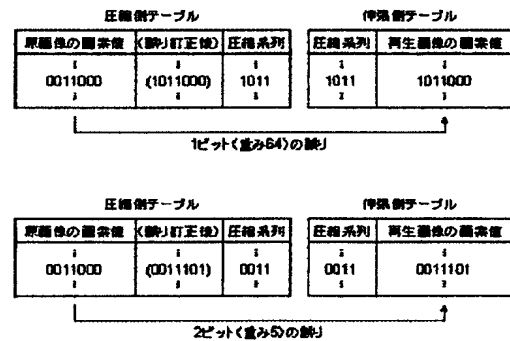
【図1】



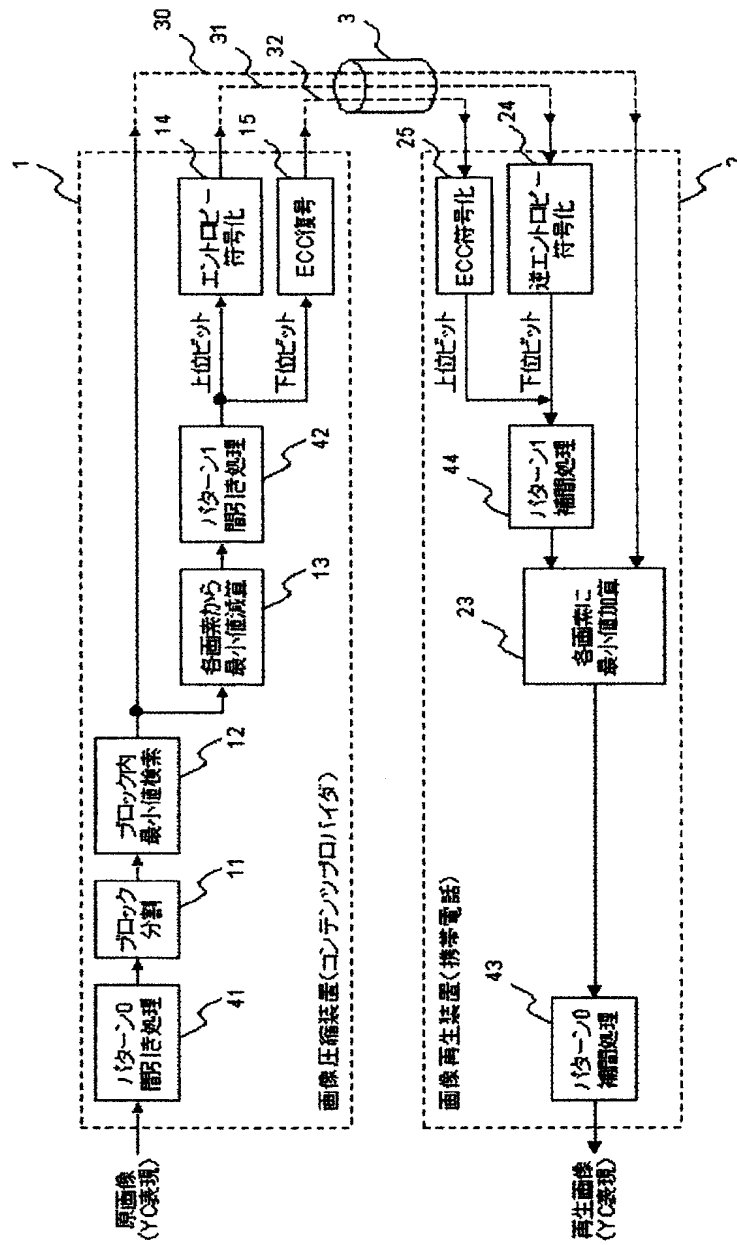
【図3】



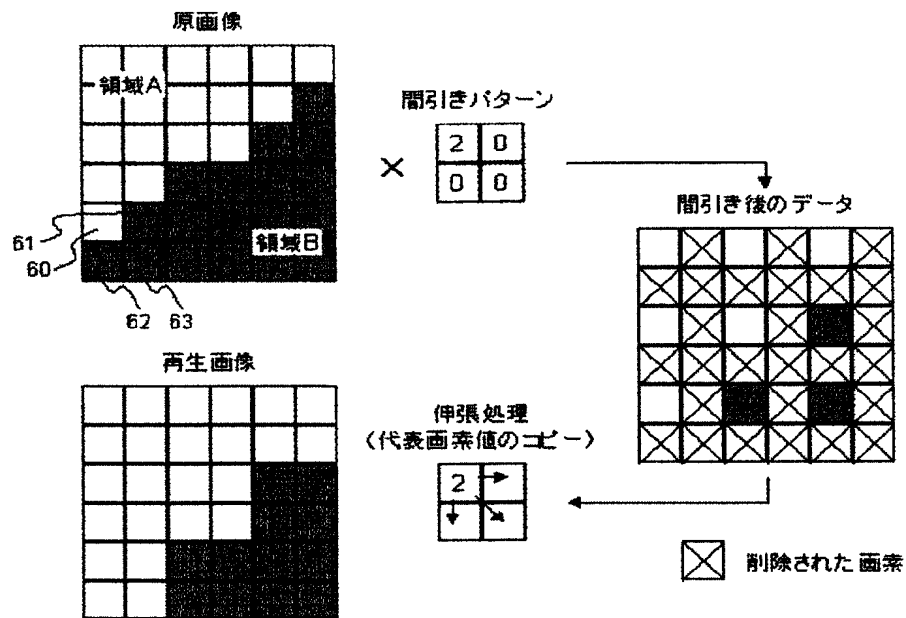
【図4】



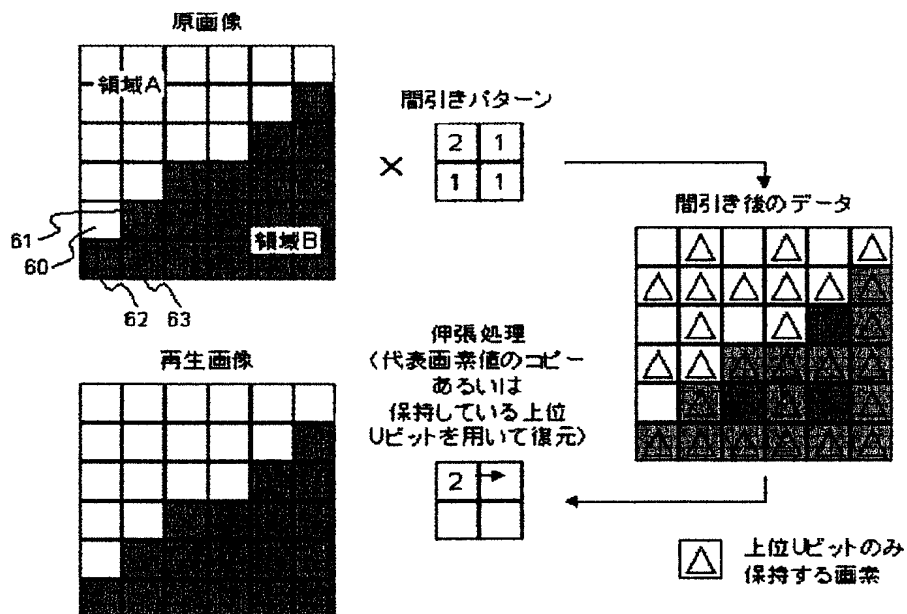
【図2】



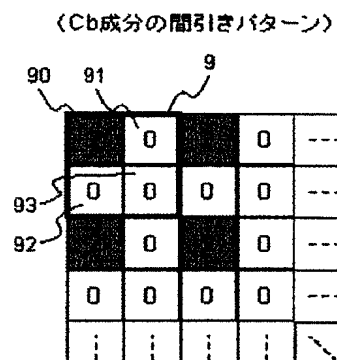
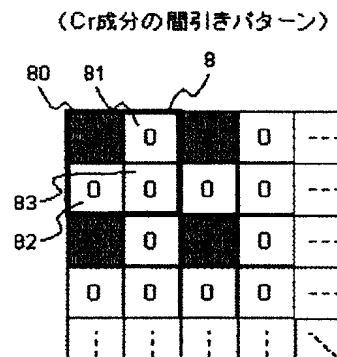
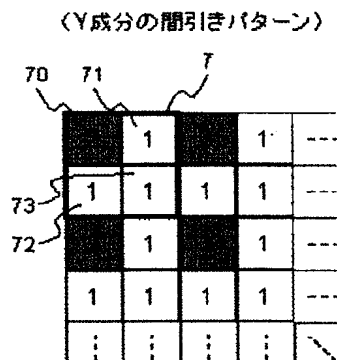
【図5】



【図6】



【図7】



【図8】

	〈保持している差分情報〉	〈補間後の差分値〉	〈補間処理の内容〉
Y成分	Y70 000001101	→ 000001101	処理不要
	Y71 00000	→ 000001101	ブロック代表画素の差分値をコピー
	Y72 0010	→ 001010000	保持情報+"1000"
	Y73 00001	→ 000011000	保持情報+"1000"

【図9】

	〈保持している情報〉	〈補間後の画素値〉	〈補間処理の内容〉
Cr成分	Cr80 0011111000	→ 0011111000	処理不要
	Cr81 [黒塗り]	→ 0011111000	ブロック代表画素値をコピー
	Cr82 [黒塗り]	→ 0011111000	ブロック代表画素値をコピー
	Cr83 [黒塗り]	→ 0011111000	ブロック代表画素値をコピー
Cb成分	Cb90 1001110001	→ 1001110001	処理不要
	Cb91 [黒塗り]	→ 1001110001	ブロック代表画素値をコピー
	Cb92 [黒塗り]	→ 1001110001	ブロック代表画素値をコピー
	Cb93 [黒塗り]	→ 1001110001	ブロック代表画素値をコピー

---

フロントページの続き

F ターム(参考) 5C059 KK15 LB11 MA00 ME02 ME05  
 PP01 PP16 RF05 SS10 SS20  
 TA06 TA08 TB08 TC02 TD15  
 UA02 UA05  
 5C078 AA09 BA44 BA57 CA45 DA01  
 DA02 DA11 DA12  
 5J064 AA01 AA02 BA08 BA09 BB08  
 BC14 BC27 BC29 BD02  
 5J065 AA01 AB03 AC02 AD05 AE02  
 AF02 AH15